2Н

УДК 004.8

Н.А. Новоселова, И.Э. Том
Объединенный институт проблем информатики
Национальной академии наук Беларуси, г. Минск
novosel@newman.bas-net.by, tom@newman.bas-net.by

Алгоритм обучения нечеткого классификатора с использованием генетических процедур

В статье рассматривается новый алгоритм обучения нечеткого классификатора, позволяющий с помощью генетических процедур обучать одновременно как базу правил, так и базу данных нечеткого классификатора (параметры функций принадлежности элементов предпосылки правила). Специально разработанные генетические процедуры позволяют параллельно оптимизировать несколько критериев, отвечающих за точность, интерпретируемость и компактность нечеткого классификатора. Сравнительное тестирование разработанного алгоритма с зарубежными аналогами на тестовом наборе данных Wine показало его преимущество в части интерпретируемости при сохранении высокой точности классификации.

Введение

В исследованиях по нечеткому моделированию актуальной является задача построения нечеткого классификатора на основе данных и соответствующего алгоритма его обучения, обеспечивающего высокую точность классификации с сохранением интерпретируемости набора нечетких правил. Существует много алгоритмов обучения, большинство из которых ориентированы на получение точной модели классификации. В последнее время разработан ряд алгоритмов, которые наряду с обеспечением точности позволяют сохранять интерпретируемость модели за счет наложения ограничений на изменение ее параметров, либо использования эвристических пост-процедур упрощения ее структуры. Такие подходы основаны на разграничении процессов оптимизации точности модели и упрощения ее структуры. В работе [1] одним из авторов статьи был предложен алгоритм обучения параметров нечеткого классификатора (НК), который решал задачу одновременной оптимизации точности и интерпретируемости НК на основе стратегии Парето-оптимальности. Данный подход позволял построить базу правил (БП) НК при неизменных изначально заданных нечетких разбиениях входных признаков. Результатом работы этого эволюционного алгоритма являлся набор недоминируемых классификаторов с различными соотношениями точности и интерпретируемости, что позволяло эксперту или специалисту в предметной области выбрать один из вариантов классификатора, приемлемый для конкретных условий.

Цель данной статьи – снять ограничения ранее разработанного алгоритма и предложить алгоритм обучения, позволяющий с помощью специальных генетических процедур обучать *параллельно* как базу правил, так и базу данных (БД) нечеткого классификатора (параметры функций принадлежности элементов предпосылки правила). В качестве оптимизационного критерия, отвечающего за простоту модели, в настоящей работе в отличие от предыдущей [1], где минимизировалось количество правил и функций принадлежности, предлагается использовать агрегированную меру сходства нечетких множеств модели. В результате применения алгоритма обучения в составе выходного набора недоминируемых классификаторов должны присутствовать экземп-

ляры, обладающие высокой точностью классификации и одновременно интерпретируемые с точки зрения различимости функций принадлежности, небольшого количества нечетких множеств в предпосылках правил и самих правил.

1 Постановка задачи

Пусть Ω – множество различных наборов нечетких правил. Каждое из правил, входящих в набор, представляет собой некоторую комбинацию лингвистических значений нечетких множеств отдельных входных признаков из максимально возможного количества комбинаций $(p_1+1)\times\ldots\times(p_n+1)$, где p_i – количество нечетких множеств, соответствующее i-му входному признаку, $i = 1,\ldots,n$. Требуется найти набор нечетких правил $S\in\Omega$, который является решением многокритериальной задачи оптимизации с критериями $F_k(S)$ ($k = 1,..,3$), где $F_1(S)$ – точность НК, измеряемая с использованием среднеквадратичной ошибки, $F_2(S)$ – интерпретируемость НК, измеряемая в виде агрегированной меры близости отдельных функций принадлежности в предпосылках правил, $F_3(S)$ – компактность НК, измеряемая количеством правил и количеством различных нечетких множеств для НК.

В процессе проведенных исследований нами был разработан алгоритм обучения НК с использованием генетических процедур, который позволяет настраивать как параметры базы данных классификатора, так и определять близкую к оптимальной структуру базы правил согласно трем критериям: точность классификации, агрегированная мера близости отдельных функций принадлежности в предпосылках правил, количество и длина правил. Специально разработанные генетические процедуры позволяют параллельно оптимизировать вышеперечисленные критерии и получать несколько недоминируемых по Парето наборов нечетких правил, которые являются компромиссными решениями и предоставляют эксперту широкие возможности по принятию окончательного решения, опираясь на знания предметной области и имеющиеся альтернативы.

2 Определение нечеткого классификатора

Нечеткий классификатор представляет собой систему нечетких правил, которые описывают m классов в имеющемся наборе исходных данных, и нечеткую систему вывода для их переработки с целью получения результата классификации [2]. Предпосылки правил задают область действия правил в n-мерном признаковом пространстве, а следствиями правил являются метки классов из множества $t\in\{C_1,\ldots,C_m\}$:

Правило R$_i$: если $x_1\in A_{i1}$ и … $x_n\in A_{in}$, то t_i, $I = 1, \ldots, M$, (1)

где M – количество правил, n – количество признаков, $x=\left[x_1,x_2,\ldots,x_n\right]^T$ – входной вектор признаков, t_i – следствие i-го правила, A_{i1}, …, A_{in} – нечеткие множества, заданные функциями принадлежности $\mu_{A_{ij}}:R\rightarrow\left[0,1\right]$; $j=1,\ldots,n$. Логическая связка «и» моделируется оператором произведения или оператором минимум в трактовке Заде, и определяет связь между отдельными элементами предпосылки. Следовательно, степень активации i-го правила рассчитывается следующим образом:

$$\beta_i(\vec{x})=\prod_{j=1}^{n}\mu_{A_{ij}}(x_j)\ \text{ или }\quad \beta_i(\vec{x})=\min_{1\le j\le n}(\mu_{A_{ij}}(x_j)),\quad i=1,2,...,M.$$

Выход классификатора определяется согласно стратегии «победитель забирает все», то есть выходом является класс, соответствующий следствию правила с наиболь-

шей степенью активации:

$$y = t_{i^*}^*, \quad i^* = \arg\max_{1 \le i \le M} (\beta_i).$$

Степень достоверности результата определяется нормализованной степенью активации правила

$$CF = \beta_{i^*} / \sum_{i}^{M} \beta_i.$$

В качестве функций принадлежности нечетких множеств отдельных признаков в нашем исследовании использовались треугольные функции, представленные формулой (2).

$$\mu_{a,b,c} : R \to [0,1], \quad \mu_{a,b,c}(x) = \begin{cases} \dfrac{x-a}{b-a}, & x \in [a,b[\\ \dfrac{c-x}{c-b}, & x \in [b,c] \\ 0, & x \in R \setminus [a,b[\cup [b,c] \end{cases}, \tag{2}$$

причем $a \le b \le c$.

При использовании более сглаженных функций принадлежности, как например гауссовы или экспоненциальные функции, в большинстве случаев НК, полученные в результате генетического обучения параметров, являются более точными только на обучающих данных.

3 Упрощение базы нечетких правил на основе меры сходства

Упрощение структуры нечеткого классификатора состоит в сокращении количества правил и в сокращении количества нечетких множеств, заданных функциями принадлежности для каждого признака. Сокращение нечетких множеств происходит за счет слияния схожих функций принадлежности для каждого из имеющихся признаков [3]. Можно выделить три типа нежелаемого сходства между нечеткими множествами, которые могут быть получены в результате выполнения автоматической процедуры построения нечеткого классификатора:

– сходство двух нечетких множеств для отдельного входного признака;

– сходство нечеткого множества с универсальным множеством U ($\mu_U(x) = 1$ $\forall x \in X$);

– сходство нечеткого множества с одноэлементным множеством.

В работе [3] предлагаются автоматические методы разрешения 1) и 2) типа сходства, вопрос удаления множеств 3) типа должен рассматриваться экспертами.

Примером меры сходства двух нечетких множеств A и B является величина:

$$S_{(A,B)} = \frac{|A \cap B|}{|A \cup B|}, \tag{3}$$

где $|\cdot|$ – определяет мощность нечеткого множества, а операторы $\cap$ и $\cup$ представляют собой операции пересечения и объединения соответственно. Если $S_{(A,B)} = 1$, то функции принадлежности A и B идентичны, $S_{(A,B)} = 0$ для неперекрывающихся функций принадлежности.

В предлагаемом нами алгоритме процедура упрощения БП нечеткого классификатора осуществляется путем объединения двух наиболее схожих нечетких множеств с последующим обновлением базы правил. Процедура повторяется до тех пор, пока мера сходства не будет ниже изначально заданного порога $\theta_S \in [0,1]$.

Если $S_{(A,B)} > \theta_S$ (нами используется $\theta_S = 0{,}6$), тогда объединение нечетких множеств A и B в новое множество C осуществляется следующим образом:

$$a_C = \min\{a_A, a_B\}$$
$$b_C = \alpha b_A + (1-\alpha) b_B ,$$
$$c_C = \max\{c_A, c_B\}$$

где коэффициент $\alpha \in [0,1]$ определяет влияние A и B на новое нечеткое множество C.

Объединение нечетких множеств сокращает количество функций принадлежности, используемых в модели, и, следовательно, улучшает интерпретируемость результатов. Если все нечеткие множества входного признака подобны универсальному множеству, или если после объединения признаку соответствует только одна функция принадлежности, то данный признак может быть исключен из модели. В результате такого упрощения в БП нечеткого классификатора могут появиться несколько правил с одинаковыми предпосылками. Избыточные правила автоматически удаляются из БП.

4 Оптимизационные критерии классификатора

В нашем исследовании рассматриваются три критерия оптимизации нечеткого классификатора:

– точность;
– интерпретируемость (простота);
– компактность.

Для каждого из критериев необходимо выбрать функцию, которая может быть вычислена в процессе оптимизации критерия.

Точность НК измеряется с использованием среднеквадратичной ошибки:

$$MSE = \frac{1}{N}\sum_{k=1}^{N}\left(y_k - \hat{y}_k\right)^2 , \qquad (4)$$

где y_k – реальное значение выходного признака для k-го входного вектора, а $\hat{y}_k$ – моделируемое значение выходного признака для k-го входного вектора, N – количество объектов данных.

Для определения второго критерия оптимизации – интерпретируемости или простоты НК – возможно использовать различные меры. В нашем исследовании мы рассматриваем меру сходства, как наиболее важную с точки зрения интерпретируемости. Значение меры сходства S между различными нечеткими множествами для каждого из входных признаков НК можно выразить следующим образом:

$$S = \max_{\substack{i,j,k \\ A_{ij} \neq B_{ik}}} S\left(A_{ij}, B_{ik}\right), i = 1,\ldots,n;\ j = 1,\ldots,M;\ k = 1,\ldots,M. \qquad (5)$$

Эта мера является агрегированной мерой сходства для нечеткого классификатора и целью ее использования в качестве критерия оптимизации является необходимость минимизировать максимальное значение сходства между нечеткими множествами для каждого входного признака.

Мерой задания третьего критерия – компактности – являются M – количество правил и L – количество различных нечетких множеств для НК. В этом случае делается предположение, что НК с малым количеством правил и нечетких множеств является компактным.

5 Реализация алгоритма обучения классификатора

5.1 Характеристики алгоритма обучения

Основными характеристиками разработанного алгоритма обучения параметров НК являются:

1) разработанный алгоритм является генетическим алгоритмом (ГА) для многокритериальной оптимизации нечеткого классификатора и позволяет за один запуск ГА получить несколько недоминируемых решений согласно Парето-стратегии;

2) определены ограничения на изменения параметров НК в процессе оптимизации, которые учитывают специфические знания предметной области. Генетические процедуры инициализации и рекомбинации особей популяции учитывают эти ограничения;

3) алгоритм обучения выполняет поиск оптимальной структуры НК в пространстве, охватывающем НК с различным количеством правил. Каждая особь популяции содержит различное количество правил в интервале $\left[1, \max\right]$, где $\max$ задается экспертом. Нечеткие множества в предпосылках правил кодируются действительными числами;

4) начальная популяция ГА генерируется случайным образом и равномерно распределена в границах поискового пространства, которое определяется обучающими данными и заданными ограничениями;

5) поиск решений осуществляется среди НК с упрощенными базами правил, то есть все особи популяции предварительно упрощаются (после инициализации и рекомбинации) и имеют значение сходства в интервале $\left[0, 0.6\right]$;

6) генетические процедуры отбора и замены особей популяции выполняются с использованием схемы предварительного отбора [4], которая является одновременно процедурой формирования ниш и стратегией элитизма ГА [5]. Кроме того, используется эвристический подход для сохранения разнообразия в количестве наборов правил особей ГА. Выживаемость особей популяции базируется на концепции Парето-оптимальности;

7) используемые генетические процедуры рекомбинации выполняются для нескольких уровней структуры НК: уровень базы правил; уровень отдельного правила; уровень параметров функций принадлежности.

Общая схема работы предложенного алгоритма представлена на рис. 1.

5.2 Представление структуры классификатора в особи генетического алгоритма и инициализация начальной популяции

Параметры каждого нечеткого классификатора (решение оптимизационной задачи) кодируются в хромосоме $\vec{s}_l$, $l = 1,\ldots,L$ (L – размер популяции), как последовательность элементов, описывающих нечеткие множества предпосылок правил. Классификатор с M нечеткими правилами кодируется следующим образом:
$\vec{s}_l = \left(ant_1,\ldots,ant_M\right)$, где строка $ant_i = \left(a_{i1}, b_{i1}, c_{i1}, \ldots, a_{in}, b_{in}, c_{in}\right)$ содержит параметры нечетких множеств A_{ij}, $j = 1,\ldots,n$ – предпосылки правила i.

При генерации начальной популяции используются несколько ограничений:

1) ограничение согласно семантике нечеткого множества. Согласно этому ограничению каждое нечеткое множество A_{ij} $(i = 1,\ldots,M;\ j = 1,\ldots,n)$ может быть представлено тремя действительными значениями $a_{ij}, b_{ij}, c_{ij} \in \left[l_j, u_j\right]$, где $a_{ij} \le b_{ij} \le c_{ij}$;

2) пара соседних функций принадлежности отдельных признаков должна удовлетворять условию $a_R \le c_L$, где L и R обозначают левое и правое нечеткое множество соответственно;

3) количество нечетких правил, закодированных в каждой популяции ГА, лежит в интервале $\left[1, \max\right]$.

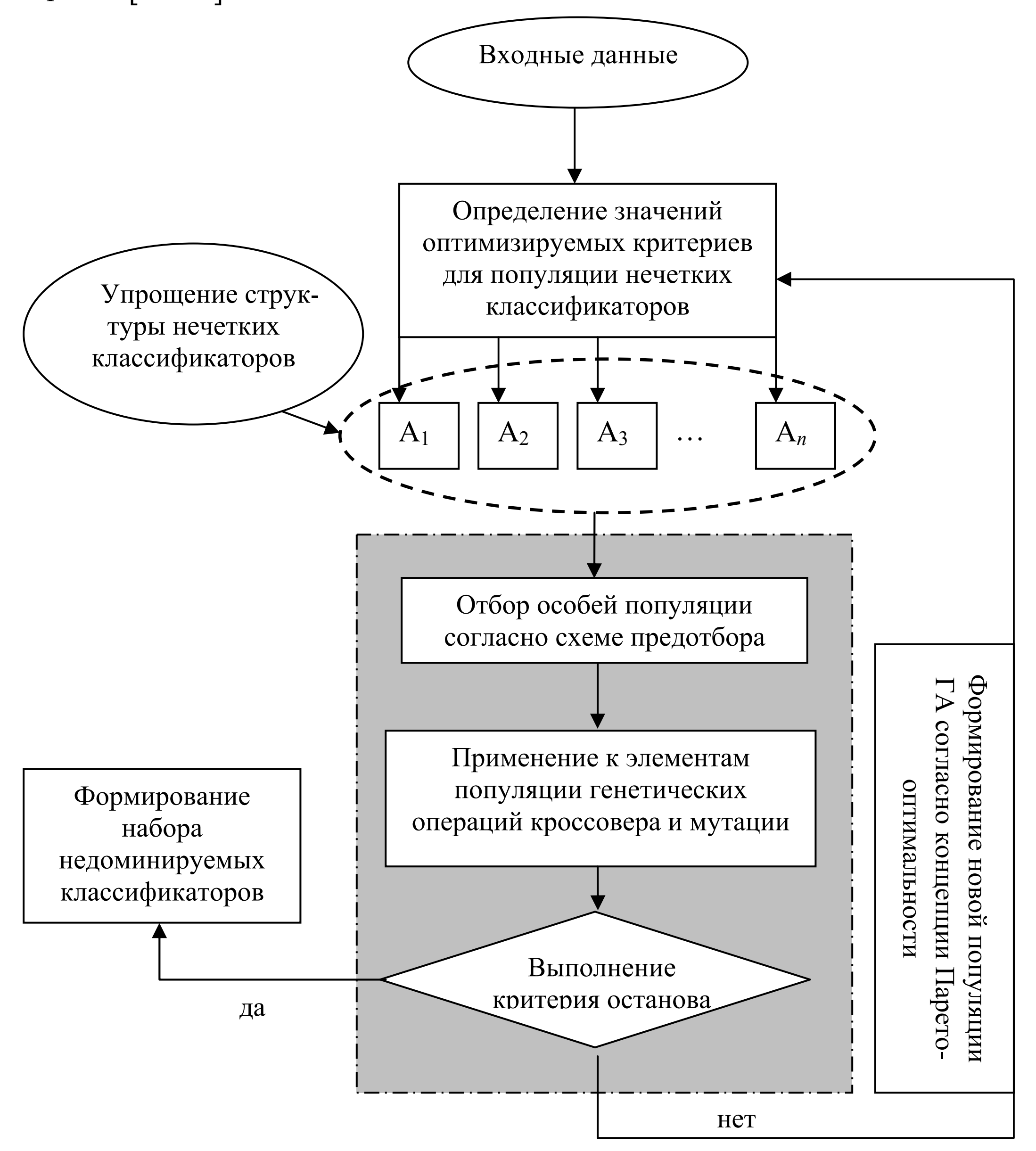


Рисунок 1 – Общая схема работы алгоритма обучения нечеткого классификатора с использованием генетических процедур

Процедура инициализации начальной популяции осуществляется следующим образом:

1) начальная популяция ГА генерируется случайным образом так, что количество особей с M закодированными правилами для каждого значения $M \in [1, \max]$ лежит в интервале [*minNS, maxNS*], $0 \leq minNS \leq \frac{L}{\max} \leq maxNS \leq L$, где L – размер популяции, *minNS* – минимальный размер ниши особей популяции; *minNS* – максимальный размер ниши особей популяции;

2) для генерации каждой особи, кодирующей M правил: для каждой треугольной функции принадлежности, задающей нечеткое множество A_{ij} $(i = 1,\ldots,M;\ j = 1,\ldots,n)$, случайным образом в интервале $\left[l_j, u_j\right]$ генерируются три действительных значения, которые далее сортируются по возрастанию для удовлетворения ограничения $a_{ij} \leq b_{ij} \leq c_{ij}$;

3) для каждой особи популяции выполняется процедура упрощения базы правил сгенерированного нечеткого классификатора.

5.3 Процедура отбора и замены особей популяции

Для отбора особей популяции на каждой последующей генерации используется схема предварительного отбора [4]. Процедура отбора и замены особей популяции выполняется следующим образом:

1) на каждой итерации ГА случайным образом отбираются две особи популяции;

2) для отобранных особей выполняются n операций скрещивания и для каждой из полученных дочерних особей выполняется операция мутации. Таким образом, получается $2 \times n$ особей;

3) наилучшая из первых n особей замещает первую родительскую особь, и наилучшая из последующих n особей замещает вторую родительскую особь только, если:

– потомок лучше родительской особи;

– количество правил, закодированных в дочерней особи, равно количеству правил родительской особи или величина ниши для родительской особи больше, чем *minNS*, и величина ниши дочерней особи меньше, чем *maxNS*.

Особь I лучше, чем другая особь J, если I доминирует J. Наилучшая особь I из некоторой коллекции особей – это особь, которая не доминируется ни одной из особей J коллекции. Величина ниши особи I – это количество особей в популяции, кодирующее такое же количество нечетких правил.

Таким образом, используемая схема предварительного отбора позволяет поддерживать разнородность особей популяции, так как потомок заменяет только похожего на себя родителя, а также является стратегией элитизма, так как каждая особь популяции может замещаться только лучшей особью. Для сохранения разнообразия популяции относительно количества кодированных правил используется эвристический подход, гарантирующий сохранение количества особей с M правилами для всех $M \in [1, \max]$ в интервале [*minNS, maxNS*].

5.4 Генетические процедуры скрещивания и мутации

В связи с тем, что поисковое пространство оптимизационного алгоритма достаточно большое, то для того, чтобы обеспечить эффективный поиск, используются операции рекомбинации, предназначенные для работы на различных уровнях структуры НК: уровень всего набора правил, уровень отдельных правил, уровень параметров функций принадлежности. Кроме того, после выполнения процедуры рекомбинации для каж-

дого из потомков выполняется процедура упрощения БП. В последующем описании процедур рекомбинации α – случайное число, равномерно распределенное на интервале $[0,1]$.

Процедуры рекомбинации на уровне всего набора правил:

Процедура скрещивания 1:

Пусть $I_1 = \left(R_1^1,\ldots,R_{M_1}^1\right)$ и $I_2 = \left(R_1^2,\ldots,R_{M_1}^2\right)$ – две родительские особи. Процедура скрещивания позволяет родительским особям обменяться информацией о количестве правил и об отдельных правилах каждой из них. Точки скрещивания могут располагаться только между отдельными правилами. В результате получается две дочерних особи: $I_3 = \left(R_1^1,\ldots,R_a^1 R_{b+1}^2,\ldots,R_{M_2}^2\right)$ и $I_4 = \left(R_{a+1}^1,\ldots,R_{M_1}^1 R_1^2,\ldots,R_b^2\right)$, где

$$\begin{aligned} a &= round\left(\alpha\cdot M_1 + (1-\alpha)\cdot M_2\right) \\ b &= round\left((1-\alpha)\cdot M_1 + \alpha\cdot M_2\right) \end{aligned}. \quad (6)$$

Таким образом, количество правил дочерних особей находится в интервале $[M_1, M_2]$.

Процедура скрещивания 2:

Процедура позволяет увеличить количество правил двух дочерних особей. Первая дочерняя особь содержит все M_1 правила первой родительской особи и $\min\{\max - M_1, M_2\}$ правил второй родительской особи. Вторая дочерняя особь содержит все M_2 правила второй родительской особи и $\min\{\max - M_2, M_1\}$ правила первой родительской особи.

Процедура мутации 1:

Процедура позволяет добавить или удалить с равной вероятностью одно правило в БП. При выполнении удаления – случайным образом отобранное правило удаляется из БП. При выполнении добавления – случайным образом согласно процедуре инициализации, описанной в 5.2, генерируется новое правило, которое добавляется в БП.

Процедуры рекомбинации на уровне отдельных правил:

Процедура скрещивания 3:

Пусть $I_1 = \left(R_1^1 \ldots R_i^1 \ldots R_{M_1}^1\right)$ и $I_2 = \left(R_1^2 \ldots R_j^2 \ldots R_{M_2}^2\right)$ – две родительские особи. Процедура позволяет получить следующие две дочерние особи: $I_3 = \left(R_1^1 \ldots R_i^3 \ldots R_{M_1}^1\right)$ и $I_4 = \left(R_1^2 \ldots R_j^4 \ldots R_{M_2}^2\right)$, где

$$\begin{aligned} R_i^3 &= \alpha R_i^1 + (1-\alpha) R_j^2 \\ R_j^4 &= \alpha R_j^2 + (1-\alpha) R_i^1 \end{aligned}, \quad (7)$$

где i, j – случайные индексы из интервалов $[1, M_1]$ и $[1, M_2]$ соответственно.

Процедура скрещивания 4:

Пусть $I_1 = \left(R_1^1 \ldots R_i^1 \ldots R_{M_1}^1\right)$ и $I_2 = \left(R_1^2 \ldots R_j^2 \ldots R_{M_2}^2\right)$ – две родительские особи. Процедура позволяет получить следующие две дочерние особи: $I_3 = \left(R_1^1 \ldots R_i^3 \ldots R_{M_1}^1\right)$ и $I_4 = \left(R_1^2 \ldots R_j^4 \ldots R_{M_2}^2\right)$, где R_i^3 и R_j^4 являются результатом равномерного скрещивания.

Процедура мутации 2:

Процедура позволяет заменять случайным образом отобранное правило на новое правило, которое генерируется случайным образом согласно процедуре инициализации, описанной в пункте 5.2.

Процедуры рекомбинации на уровне параметров БП:

Процедура скрещивания 5:

Пусть $I_1 = \left(R_1^1 \dots R_i^1 \dots R_{M_1}^1\right)$ и $I_2 = \left(R_1^2 \dots R_j^2 \dots R_{M_2}^2\right)$ – две родительские особи. Процедура позволяет получить следующие две дочерние особи: $I_3 = \left(R_1^1 \dots R_i^3 \dots R_{M_1}^1\right)$ и $I_4 = \left(R_1^2 \dots R_j^4 \dots R_{M_2}^2\right)$, где правила R_i^3 и R_j^4 отобраны случайным образом и получены как результат выполнения арифметического скрещивания нечетких множеств случайным образом выбранного входного признака k:

$$\begin{matrix} a_k^3 = \alpha a_k^1 + (1-\alpha) a_k^2 & b_k^3 = \alpha b_k^1 + (1-\alpha) b_k^2 & c_k^3 = \alpha c_k^1 + (1-\alpha) c_k^2 \\ a_k^4 = \alpha a_k^2 + (1-\alpha) a_k^1 & b_k^4 = \alpha b_k^2 + (1-\alpha) b_k^1 & c_k^4 = \alpha c_k^2 + (1-\alpha) c_k^1 \end{matrix}. \qquad (8)$$

Процедура мутации 3:

Процедура позволяет изменять случайным образом выбранное нечеткое множество случайного нечеткого правила. Новое нечеткое множество генерируется случайным образом.

Процедура мутации 4:

Процедура позволяет изменять значение одного из параметров нечеткого множества a, b, или c функции принадлежности. Новое значение параметра генерируется случайным образом с учетом ограничений.

6 Оптимизационные схемы и принятие решения

В результате предварительных экспериментов было установлено следующее:

1) минимизация количества правил M оказывает отрицательное влияние на результат оптимизации. Причиной может являться то, что этот параметр не является независимой оптимизируемой переменной. В последующих экспериментах этот параметр не учитывается в оптимизации, но принимается во внимание при проведении анализа последней популяции решения задачи;

2) необходимо учитывать, что очень простая нечеткая модель не будет принята экспертом, если она не является достаточно точной. В результате проведения оптимизации по нескольким критериям промежуточные достаточно точные варианты НК часто отбрасываются из-за большого значения критерия, отвечающего за простоту модели. Таким образом, необходимо исследовать альтернативный подход с использованием ограничения g_s на значение агрегированной меры сходства, что позволит остановить минимизацию этого критерия для вариантов решений, удовлетворяющих ограничению g_s;

3) значение критерия L – количество различных нечетких множеств для НК, значительно сокращается с использованием процедуры упрощения БП нечеткого классификатора и поэтому нет необходимости определять его в качестве одного из оптимизируемых критериев.

Согласно приведенным выше результатам анализа работы предлагаемого алгоритма обучения НК нами рассматриваются две оптимизационных схемы (табл. 1).

Таблица 1 – Схемы оптимизации

Оптимизационная схема 1	минимизировать $f_1 = MSE$ и $f_2 = S$
Оптимизационная схема 2	минимизировать $f_1 = MSE$ и $f_2 = \max\{g_s, S\}$

После завершения работы алгоритма принятие окончательного решения и отбор решений осуществляется согласно следующему алгоритму:

1) определение набора $Z^* = \{z_1^*, \ldots, z_p^*\}$ недоминируемых решений по трем критериям: *минимизировать* $f_1 = MSE$ $f_2 = S$ $f_3 = M$;

2) выбор из множества Z^* наиболее точного решения z_i^*; удаление z_i^* из Z^*;

3) если решение z_i^* недостаточно точное, или множество Z^* – пустое, то завершение алгоритма отбора;

4) если решение z_i^* недостаточно простое или компактное, то переходим к шагу 2, иначе выбор решения z_i^* для вывода.

Сравнительное тестирование разработанного алгоритма с зарубежными аналогами на наборе данных Wine из международного архива тестов показало его преимущество в части интерпретируемости НК при сохранении сопоставимой точности классификации (табл. 2).

Таблица 2 – Результаты классификации набора данных Wine различными методами

Алгоритм обучения НК	Точность классификации (средняя для 10 запусков ГА)	Количество правил (средняя для 10 запусков ГА)
Алгоритм, описанный A.L. Corcoran, S. Sen	99,5 %	60
Алгоритм, описанный H. Ishibuchi, T. Nakashima	98,5%	60
Метод, описанный H. Ishibuchi, T. Nakashima, T. Murata	95,5%	10,9
Алгоритм, представленный в настоящем исследовании	**97,8%**	**8**

Заключение

В настоящей работе предложен алгоритм обучения нечеткого классификатора с использованием генетических процедур, который позволяет не только обучать БП, но и БД нечеткого классификатора путем настройки параметров функций принадлежности. Наряду с высокой эффективностью разработанного алгоритма обучения НК, о чем свидетельствует сравнительный анализ с аналогами, важным достоинством алгоритма является возможность получения нескольких классификаторов за один запуск ГА, при этом не требуются дополнительные процедуры кластеризации для определения начальной популяции особей ГА. Благодаря многокритериальной оптимизации результатом работы алгоритма являются наборы НК с различными соотношениями точности классификации и простоты структуры. В качестве критерия интерпретируемости НК используется агрегированная мера сходства нечетких множеств. Особи ГА имеют переменную длину и кодируют параметры предпосылок базы нечетких правил НК, следствия нечетких правил рассчитываются автоматически на каждой итерации ГА. Разработанные генетические процедуры инициализации, отбора, рекомбинации особей ГА позволяют эффективно искать решения в поисковом пространстве, охватывающем НК с различным количеством правил.

Литература

1. Новоселова Н.А. Построение нечеткой модели классификации с использованием многокритериального генетического алгоритма / Н.А. Новоселова // Искусственный интеллект. – 2006. – № 3. – С. 613-622.
2. Zimmermann H.J. Fuzzy Set Theory and its Applications / H.J. Zimmermann. – Boston, Dordrecht, Lancaster : Kluwer Academic Publishers, 1996.
3. Setnes M. Similarity measures in fuzzy rule base simplification / M. Setnes, R. Babuˇska, U. Kaymak, and H.R. van Nauta Lemke // IEEE Trans. Systems, Man and Cybernetics. – Part B. – 1998. – Vol. 28. – P. 376-386.
4. Goldberg D.E. Genetic Algorithms in Search, Optimization, And Machine Learning / Goldberg D.E. – Addison-Wesley, 1989.
5. Michalewicz Z. Genetic Algorithms + Data Structures = Evolution Programs / Michalewicz Z. – [2nd ed.] New York : Springer Verlag, 1994.

Н.А. Новосьолова, І.Э. Том

Алгоритм навчання нечіткого класифікатора з використанням генетичних процедур

Розглядається новий алгоритм навчання нечіткого класифікатора, що дозволяє за допомогою генетичних процедур навчати одночасно як базу правил, так і базу даних нечіткого класифікатора (параметри функцій приналежності елементів передумови правила). Спеціально розроблені генетичні процедури дозволяють паралельно оптимізувати кілька критеріїв, відповідальних за точність, інтерпретованість і компактність нечіткого класифікатора. Порівняльне тестування розробленого алгоритму із закордонними аналогами на тестовому наборі даних Wine показало його перевагу у частині інтерпретованості при збереженні високої точності класифікації.

N.A. Novoselova, I.E. Tom

Learning Algorithm of Fuzzy Classifier with Genetic Procedures

In the paper the new learning algorithm of fuzzy classifier (FK) is proposed, which uses the genetic procedures to simultaneously adjust both the rule base and data base (the parameters of membership function of rule premises) of FK. The specially developed genetic procedures permit to optimize in parallel several criteria, responsible for classification accuracy, simplicity and compactness of fuzzy classifier. The comparative analysis of developed algorithm on the testing dataset Wine shows its advantage over foreign analogs according to interpretability of results preserving the high classification accuracy.

Статья поступила в редакцию 26.05.2010.